

 <p>دانشگاه صنعتی خواجه نصیرالدین طوسی K. N. TOOSI UNIVERSITY OF TECHNOLOGY</p>	<p>دکتر نیک انجام - دکتر نصیحت کن</p>	<p>امتحان پایان ترم میانی کامپیوتر</p>
<p>دیماه ۱۳۹۵</p>	<p>شماره دانشجویی:</p>	<p>نام:</p>

توجه: برنامه هایی که می نویسید باید خوانا و مرتب باشند. مخصوصا تو رفتگی ها (indentation) باید رعایت شده باشد.

سوال ۱ (۱۲ نمره)

ماتریس مربعی A پادمتقارن (skew-symmetric) است اگر داشته باشیم $A^T = -A$. مثال:

$$\begin{bmatrix} 0 & 2 & -1 \\ -2 & 0 & -4 \\ 1 & 4 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & -2 & -4 & 16 & 8 \\ 2 & 0 & 7 & -6 & 1 \\ 4 & -7 & 0 & -4 & 11 \\ -16 & 6 & 4 & 0 & -8 \\ -8 & -1 & -11 & 8 & 0 \end{bmatrix}$$

تابعی بنویسید که یک ماتریس N در N را به صورت یک آرایه دو بعدی بگیرد و اگر ماتریس پادمتقارن بود عدد ۱ و در غیر این صورت عدد صفر را برگرداند. (فرض کنید N عددی ثابت است که بوسیله `#define` تعریف شده.)

```
int is_skew_symmetric(int A[][N]) {
    int i,j;

    for (i=0; i<N; i++) {
        for (j=0; j<N; j++) {
            if (A[i][j] != -A[j][i])
                return 0;
        }
    }

    return 1;
}
```

سوال ۲ (۱۵ نمره)

الف) تابع زیر چه کاری انجام می دهد؟ (۷ نمره)

```
void rc(char d[], char s[]) {
    int n,i;

    for (n=0; s[n] != '\0' ; n++)
        ;

    for (i=0; i < n; i++)
        d[i] = s[n-i-1];

    d[n] = '\0';
}
```

معکوس رشته s را در رشته d کپی می کند.

ب) تابع بالا را با استفاده از اشاره گر pointer (بدون استفاده از عملگر براکت []) بازنویسی کنید. کد نهایی را در کادر زیر بنویسید. (۸ نمره)

```
void rcl(char *d, char *s) {
    int n;

    for (n=0; *s != '\0' ; n++,s++)
        ;

    for (s--; n > 0; n--,s--,d++)
        *d = *s;

    *d = '\0';
}
```

راه حل جایگزین:

```
void rc2(char *d, char *s) {
    int n,i;

    for (n=0; *(s+n) != '\0' ; n++)
        ;

    for (i=0; i < n; i++)
        *(d+i) = *(s+n-i-1);

    *(d+n) = '\0';
}
```

سوال ۳ (۱۵ نمره)

در کد زیر دو تابع `sw` و `ins` تعریف شده اند. تابع `ins` یک آرایه از اعداد `double` به نام `arr` و همچنین یک عدد `n` به عنوان طول آرایه را دریافت می کند.

```
void sw(double arr[], int i, int j) {
    double tmp = arr[i];
    arr[i] = arr[j];
    arr[j] = tmp;
}

void ins(double arr[], int n) {
    int i,k;
    for (k = 1; k < n; k++)
        for (i = k; i > 0 && arr[i] > arr[i-1]; i--)
            sw(arr,i,i-1);
}
```

1. تابع `sw` چه کاری انجام می دهد؟ (۲ نمره)

جای اعضای `i` ام و `j` ام آرایه `arr` را با هم عوض می کند.

2. تابع `ins` چه کاری انجام می دهد؟ (۵ نمره)

آرایه را به ترتیب نزولی (از بزرگ به کوچک) مرتب می کند

3. طرز کار تابع `ins` را توضیح دهید؟ (۶ نمره)

در این تابع در حلقه اول (حلقه بیرونی) مقدار `k` از ۱ تا `n-1` تغییر می کند. الگوریتم طوری نوشته شده که در هر تکرار از حلقه اول و قبل از شروع اجرای حلقه دوم (حلقه درونی) اعضای شماره 0 تا `k-1` از آرایه `arr` مرتب شده هستند. پس از اجرای حلقه دوم اعضای 0 تا `k` آرایه `arr` مرتب شده خواهند بود. حلقه دوم `a[k]` را در محل مناسب خود در بین اعضای شماره 0 تا `k-1` از آرایه `arr` قرار می دهد به صورتیکه اعضای 0 تا `k` در نهایت مرتب شده شوند. بدین ترتیب وقتی `k` مقادیر 1 تا `n-1` را می گیرد و در نهایت مقدار آن برابر `n` می شود تمام لیست مرتب شده خواهد بود.

(صرفاً جهت اطلاع نام این الگوریتم `insertion sort` است.)

4. اگر تابع `ins` را بر روی یک آرایه صدا بزینم ولی آرگومان `n` را کمتر از تعداد اعضای واقعی آرایه بدهیم چه اتفاقی می افتد؟ (۲ نمره)

`n` عضو اول آرایه مرتب می شود.

سوال ۴ (۲۰ نمره)

فرض کنید در ماشین مورد نظر ما هر متغیر `int` چهار بایت از حافظه را اشغال می کند. متغیر `a` به صورت آرایه ای از `int` تعریف شده. فرض کنید آدرس ابتدای آرایه `a` (آدرس اولین خانه `a`) در حافظه برابر با 1020 باشد. شکل زیر آرایه `a` را به همراه محتویات اعضای آن نشان می دهد:

1020

8	-13	9090	1111	-102	33	25	80	-10	200
---	-----	------	------	------	----	----	----	-----	-----

در قطعه کد زیر آرایه `a` و همچنین دو اشاره گر `p` و `q` تعریف و مقدار دهی شده اند:

```
int a[10] = {8, -13, 9090, 1111, -102, 33, 25, 80, -10, 200};
int *p = a;
int *q = p+4;
```

الف) مقدار هر کدام از عبارات زیر را بنویسید (۱۲ نمره)

<code>a[4]</code>	-102	<code>*p</code>	8	<code>p</code>	1020	<code>*(p+4)</code>	-102
<code>p[4]</code>	-102	<code>*q</code>	-102	<code>q</code>	1036	<code>*(q-2)</code>	9090
<code>q[4]</code>	-10	<code>&q[0]</code>	1036	<code>p+1</code>	1024	<code>q-p</code>	4
<code>q[0]</code>	-102	<code>&p[7]</code>	1048	<code>q-1</code>	1032	<code>p-q</code>	-4

ب) در زیر دو برنامه می بینید که از توابع بازگشتی استفاده کرده اند. اگر هر کدام از این توابع را روی آرایه `a` (که در بالا تعریف شده) صدا بزنیم، خروجی آن ها را در کادر پایین هر کد بنویسید. (۸ نمره)

<pre>void p1(int *a, int n) { if (n == 0) return; printf("%d, ", a[0]); p1(a+1, n-1); } int main() { int a[10] = {8, -13, 9090, 1111, -102, 33, 25, 80, -10, 200}; p1(a, 10); }</pre>	<pre>void p2(int *a, int n) { if (n == 0) return; p2(a+1, n-1); printf("%d, ", a[0]); } int main() { int a[10] = {8, -13, 9090, 1111, -102, 33, 25, 80, -10, 200}; p2(a, 10); }</pre>
8, -13, 9090, 1111, -102, 33, 25, 80, -10, 200,	200, -10, 80, 25, 33, -102, 1111, 9090, -13, 8,

سوال ۵- (۱۳ نمره)

در کد زیر یک ساختار به نام Student تعریف شده که شامل نام، نام خانوادگی و شماره دانشجویی است. همچنین در تابع main آرایه ای از این ساختار به نام students ساخته شده و دو متغیر s و ps تعریف شده اند:

```
struct Student {
    char firstName[20];
    char lastName[20];
    unsigned int id;
};

int main() {

    struct Student students[] = {
        {"Mohsen", "Ahadi", 9501235},
        {"Mona", "Salari", 9509574},
        {"Hoda", "Rahmati", 9507494},
        {"Ramin", "Birang", 9518978}
    };

    struct Student s;
    struct Student *ps;
```

الف: با توجه به کد بالا، بنویسید هر کدام از قطعه کد های زیر چه چیزی چاپ می کنند. اگر کد مربوطه خطای کامپایل دارد علت آن را ذکر کنید. (۱۵ نمره)

تابع puts یک رشته را می گیرد و آن را چاپ می کند. اپراتور -> روی یک اشاره گر به ساختار اعمال می شود و فیلد مربوطه ساختار را بر می گرداند. بنابراین خطوط زیر معادل می باشند:

ps->id
(*ps).id

<pre>s = students; printf("%d\n", s.id);</pre>	خطای کامپایل، students که آرایه ای از ساختارهاست به s که از نوع ساختار است انتصاب داده شده.
<pre>s = students[1]; s.id++; printf("%d\n", s.id);</pre>	9509575
<pre>s = &students[0]; ps = *s; puts(ps->firstName);</pre>	خطای کامپایل &students[0] از نوع اشاره گر به ساختار است به s که از نوع ساختار است انتصاب داده شده.
<pre>ps = &students[3]; ps--; puts(ps->firstName);</pre>	Hoda

<pre>s = students[0]; ps = &(s+1); puts(ps->firstName);</pre>	<p>خطای کامپایل: s+1 غلط است. چون s از نوع ساختار است و با عدد 1 از نوع int جمع شده. جمع ساختار با int تعریف نشده.</p>
<pre>ps = students; puts((ps+2)->lastName);</pre>	Rahmati
<pre>ps = &students[0]; puts(ps.lastName);</pre>	<p>خطای کامپایل: ps از نوع اشاره گر به ساختار است. بنابراین عملگر (dot) برای آن تعریف نشده.</p>
<pre>ps = students; ps+=3; printf("%d\n", ps.id);</pre>	<p>خطای کامپایل: ps از نوع اشاره گر به ساختار است. بنابراین عملگر (dot) برای آن تعریف نشده.</p>
<pre>ps = students; ps += 1; s = *ps; printf("%d\n", s.id);</pre>	9509574
<pre>printf("%d\n", students[2]->id);</pre>	<p>خطای کامپایل: students[2] از نوع ساختار است و عملگر -> برای ساختار تعریف نشده (هر چند برای اشاره گر به ساختار تعریف شده است)</p>
<pre>ps = students; puts((*ps+2).lastName);</pre>	Rahmati
<pre>ps = students; ps += 2; char *t = (ps+1)->firstName; t += 4; putchar(*t);</pre>	n
<pre>putchar(students[2][3]);</pre>	<p>خطای کامپایل: students[2] از نوع ساختار است (نه آرایه است و نه اشاره گر). بنابراین students[2][3] معنا ندارد.</p>

سوال ۶- (۲۵ نمره)

در دانشگاه خواجه نصیر زمان ورود و خروج کارمندان ثبت می شود. فرض کنید فایل به نام `logbook.txt` وجود دارد که هر بار که کارمندی وارد دانشگاه می شود یا از آن خارج می گردد یک خط به انتهای آن اضافه می شود (بنابر این خط های فایل به ترتیب زمانی مرتب شده هستند). در زیر می توانید مثالی از محتویات فایل `logbook.txt` را ببینید:

I	4253	1395	9	12	7	20
I	4485	1395	9	12	8	11
O	4253	1395	9	12	16	31
O	4485	1395	9	12	16	46
I	4253	1395	9	15	7	0
O	4253	1395	9	15	8	6
I	4253	1395	9	15	9	14
I	4788	1395	9	15	9	31
O	4253	1395	9	12	18	20
O	4788	1395	9	12	20	40
I	4253	1395	10	27	8	6

همانطور که عنوان شد هر سطر فایل مربوط به ورود یا خروج یکی از کارمندان است. هر سطر شامل ۷ فیلد است. فیلد اول یا I است یا O که نشان می دهد کارمند وارد دانشگاه شده یا از آن خارج شده. فیلد دوم شناسه کارمندی است که مختص هر کارمند می باشد. پنج فیلد بعدی به ترتیب سال، ماه، روز، ساعت و دقیقه می باشد. مثلا سطر اول فایل بالا نشان می دهد که کارمند با شناسه 4253 در ۱۲ آذر ۱۳۹۵ ساعت ۷ و ۲۰ دقیقه وارد دانشگاه شده.

الف) تابعی بنویسید که تاریخ امروز را بگیرد و شناسه کاربرانی را که هم اکنون در دانشگاه حضور دارند را چاپ کند: (۱۰ نمره)

```

int print_inside(int year, int month, int day, int id) {
    FILE *f = fopen("logbook.txt", "r");
    char state = 'i';
    char iol;
    int hour, min;
    int id1, year1, month1, day1, hour1, min1;

    do {
        fscanf(f, "%c %d %d %d %d %d %d ", &iol, &id1, &year1, &month1,
            &day1, &hour1, &min1);

        if (id1 == id && year1 == year && month1 == month && day1 == day) {
            state = iol;
            hour = hour1;
            min = min1;
        }

    } while (! feof(f));

    if (state == 'I') {
        printf("%02d:%02d\n",hour,min);
    }

    fclose(f);
}

```

ب) تابعی بنویسید که ماه و سال و همچنین شناسه یک کاربر خاص را بگیرد و میزان حضور کارمند در دانشگاه را در ماه مربوطه (به ساعت و دقیقه) چاپ کند: (۱۵ نمره)


```
int print_presence_time(int year, int month, int id) {
    FILE *f = fopen("logbook.txt", "r");
    char io1;
    int id1, year1, month1, day1, hour1, min1;
    int hour0, min0;

    int t = 0; // time in minutes
    do {
        fscanf(f, "%c %d %d %d %d %d %d ", &io1, &id1, &year1,
            &month1, &day1, &hour1, &min1);

        if (id1 == id && year1 == year && month1 == month) {
            if (io1 == 'I') {
                hour0 = hour1;
                min0 = min1;
            } else { // if (io1 == 'O')
                t += (hour1-hour0)*60 + (min1-min0);
            }
        }

    } while (! feof(f));

    printf("%02d:%02d\n", t/60, t%60);

    fclose(f);
}
```

سوال جایزه ای (۱۵ نمره): ماتریس U متعامد (orthogonal) نامیده می شود اگر معکوس آن برابر با ترانزپوز آن باشد، به عبارت دیگر

$$U^T U = I$$

که I ماتریس یکه است. تابعی بنویسید که یک ماتریس N در N را بگیرد و در صورت متعامد بودن یک و در غیر این صورت صفر برگرداند.

```
int is_orthogonal(double U[][N]) {
    int i,j,k;
    double Iij; // ij-th element of I
    double Aij; // ij-th element of A = U^T * U

    for (i=0; i<N; i++) {
        for (j=0; j<N; j++) {
            if (i == j)
                Iij = 1;
            else
                Iij = 0;

            Aij = 0;
            for (int k = 0; k < N; k++)
                Aij += U[k][i] * U[k][j];

            if (Aij != Iij)
                return 0;

        }
    }

    return 1;
}
```

Hint: For each i,j the variable A_{ij} will eventually be equal to the ij -th element of $A = U^T U$. Notice that A_{ij} is the product of the i -th row of U^T by the j -th column of U . This is equal to the inner product of the i -th column of U by the j -th column of U . Thus:

$$A_{ij} = \sum_{k=0}^{N-1} U_{ki} U_{kj}.$$